



Universidad Nacional Autónoma de México

Facultad de Ingeniería

Ingeniería en Computación

Ingeniería de Software

Grupo 04

Ing. Orlando Zaldívar Zamorategui

**TUTORIAL IS 2025-2 1. Introducción a la Ingeniería de Software.
Definición de
Ingeniería de Software. (CUESTIONARIO 'RESPUESTAS')**

EQUIPO 2

**Benítez Guerrero Alexis
Cabrera Marcos Sergio Ivan
Moreno Muñiz Luis Gerardo**

Ciudad Universitaria, Ciudad de México, a 27 / 05 / 2025

RESPUESTAS DEL CUESTIONARIO

1. ¿Cómo define el IEEE el software?

- A) Como el conjunto de programas que especifican las instrucciones que debe seguir el sistema.
- B) Como la documentación y procedimientos necesarios para el desarrollo de hardware.
- C) **Como el conjunto de programas, procedimientos, reglas, documentación y datos asociados de un sistema de cómputo.**
- D) Como el conjunto de aplicaciones orientadas al usuario.

Referencia: Primer párrafo, definición formal de software según el IEEE.

2. ¿Cuál de las siguientes es una característica principal del software de sistemas?

- A) Está diseñado para calcular simulaciones avanzadas.
- B) Proporciona funcionalidad específica y general para consumidores.
- C) **Optimiza el acceso y la eficiencia de otros programas y usuarios.**
- D) Especifica las reglas de negocio en aplicaciones empresariales.

Referencia: Sección sobre clasificación de software, categoría "Software de sistemas".

3. ¿Cuál de las siguientes es una ventaja del software en la vida cotidiana?

- A) Dificultad para medir el progreso.
- B) **Expansión de tecnologías de comunicación.**
- C) Costos de desarrollo elevados.
- D) Complejidad en las estimaciones de tiempo.

Referencia: Ventajas del software en nuestra vida cotidiana, punto sobre la expansión de tecnologías existentes.

4. ¿Qué tipo de software se utiliza para cálculos complejos y simulaciones en áreas científicas?

- A) Software de sistemas.
- B) Aplicaciones web.
- C) **Software científico y de ingeniería.**
- D) Software de línea de productos.

Referencia: Sección de clasificación de software, categoría "Software científico y de ingeniería".

5. ¿Cuál es una de las principales desventajas del software?

- A) La imposibilidad de realizar pruebas para verificar su funcionamiento.
- B) El reemplazo constante de componentes defectuosos.
- C) **Los altos costos de mantenimiento.**
- D) Su corta vida útil comparada con el hardware.

Referencia: Desventajas del software y sus desafíos, punto sobre elevados costos de mantenimiento.

6. ¿Qué mide la mantenibilidad como factor de calidad del software?

- A) La facilidad para modificar el software.
- B) La ausencia de fallos durante el uso.
- C) La capacidad para adaptarse a plataformas distintas.
- D) **La facilidad para corregir problemas en el software.**

Referencia: Factores de calidad del software, mantenibilidad.

7. ¿Cómo se define un producto de software genérico?

- A) Un software solicitado por un cliente específico.
- B) Un software desarrollado a medida para una organización.
- C) **Un sistema desarrollado y comercializado en el mercado abierto.**
- D) Un sistema personalizado para sistemas empresariales.

Referencia: Sección sobre productos de software, definición de productos genéricos.

8. ¿Qué implica la corrección como factor de calidad en el software?

- A) Medir la eficiencia de los recursos utilizados.
- B) **Determinar si cumple con sus especificaciones.**
- C) Evaluar el esfuerzo para aprender a usar el software.
- D) Determinar la capacidad de interoperabilidad.

Referencia: Factores de calidad del software, corrección.

9. ¿Cuál es un ejemplo de software incrustado?

- A) Un sistema operativo de uso general.
- B) Un programa de diseño gráfico.
- C) **El software de control en un electrodoméstico.**
- D) Un software ERP para empresas.

Referencia: Sección de clasificación de software, categoría "Software incrustado o empotrado".

10. ¿Qué mide la facilidad de prueba en el software?

- A) El esfuerzo para modificar el software.
- B) **La facilidad para verificar su corrección o fiabilidad.**
- C) La posibilidad de adaptarlo a otras plataformas.
- D) La facilidad para documentar los cambios.

Referencia: Factores de calidad del software, facilidad de prueba.

11. ¿Qué caracteriza al software de aplicación?

- A) Es un software orientado exclusivamente a desarrolladores.
- B) Se enfoca en optimizar el acceso y la eficiencia de recursos.
- C) **Atiende necesidades específicas de individuos u organizaciones.**
- D) Es utilizado principalmente en simulaciones científicas.

Referencia: Sección de clasificación de software, categoría "Software de aplicación".

12. ¿Cuál es un desafío común en el desarrollo de software?

- A) La facilidad para medir su calidad de manera directa.
- B) La interacción constante con componentes de hardware.
- C) La disponibilidad de componentes estandarizados.
- D) **La capacidad de probar todas las funciones del software.**

Referencia: Desventajas del software y sus desafíos, sobre la dificultad de garantizar calidad.

13. ¿Cuál es un ejemplo de una aplicación web?

- A) Un sistema operativo.
- B) Un programa de edición de imágenes.
- C) **Una aplicación bancaria en línea.**
- D) Un programa de simulación meteorológica.

Referencia: Sección de clasificación de software, categoría "Aplicaciones web".

14. ¿Qué es la flexibilidad en términos de calidad de software?

- A) La capacidad de adaptarse a otras plataformas.
- B) **La facilidad para modificar el software en el mantenimiento adaptativo.**
- C) La capacidad de realizar tareas en tiempo real.
- D) La facilidad de reusar partes del software en desarrollos futuros.

Referencia: Factores de calidad del software, flexibilidad.

15. ¿Qué función tiene el software de línea de productos?

- A) Controla el funcionamiento en dispositivos empujados.
- B) **Proporciona funcionalidad generalizada para una amplia variedad de usuarios.**
- C) Ejecuta simulaciones avanzadas y cálculos complejos.
- D) Sirve como base para otros desarrolladores de aplicaciones.

Referencia: Sección de clasificación de software, categoría "Software de línea de productos".

16. ¿Qué es la ingeniería de software?

- A) Solo programación de sistemas.
- B) Desarrollo de software sin documentación.
- C) **Diseño, desarrollo y entrega de software según los requisitos del cliente.**
- D) Mantenimiento de sistemas operativos

Referencia: Definición de ingeniería de software.

17. ¿Quién es el responsable de supervisar el proyecto de ingeniería de software en su totalidad?

- A) Cliente.
- B) Programador.
- C) **Project Manager.**
- D) End User.

Referencia: Tabla de roles y responsabilidades.

18. Según la definición de Bauer (1972), ¿qué busca lograr la ingeniería de software?

- A) Un software caro y exclusivo.
- B) **Un software eficiente y económico.**
- C) Un sistema operativo nuevo.
- D) Un software que requiera pocas actualizaciones.

Referencia: Definiciones de ingeniería de software.

19. ¿Qué función desempeña el "System Architect"?

- A) Escribe el código del sistema.
- B) Realiza pruebas de software.
- C) **Diseña el sistema y establece cómo interactúan sus componentes.**
- D) Entrena a los usuarios finales.

Referencia: Tabla de roles y responsabilidades.

20. ¿Qué significa el principio de "Modularidad" en ingeniería de software?

- A) **Dividir el software en componentes pequeños y manejables.**
- B) Integrar todo el software en un único módulo.
- C) Evitar el uso de funciones y procedimientos.
- D) Aumentar el tamaño del software.

Referencia: Principios esenciales de ingeniería de software.

21. ¿Cuál de las siguientes fases NO es parte del proceso de desarrollo de software?

- A) Análisis de requisitos.
- B) Implementación.
- C) **Distribución de hardware.**
- D) Evolución.

Referencia: Etapas del proceso de desarrollo de software.

22. ¿Qué describe el principio de "Anticipación al cambio"?

- A) Mejorar la eficiencia del sistema.
- B) **Planificar para manejar posibles cambios en el software.**
- C) Mantener un sistema sin cambios.
- D) Evitar actualizaciones.

Referencia: Principios esenciales de ingeniería de software.

23. ¿Cuál es la función principal de un "Tester"?

- A) Escribir código para el sistema.
- B) Diseñar la arquitectura del sistema.
- C) **Probar el sistema de acuerdo con los requisitos del cliente.**
- D) Administrar al equipo de desarrollo

Referencia: Tabla de roles y responsabilidades.

24. ¿Qué etapa se realiza al inicio de un proyecto de software?

- A) Evolución.
- B) Codificación.
- C) **Definición inicial.**
- D) Implementación

Referencia: Etapas del proceso de desarrollo de software.

25. ¿Cuál es el objetivo de la "Gestión de la configuración del software"?

- A) Codificar el software.
- B) **Controlar los cambios en el software.**
- C) Implementar nuevas funciones sin errores.
- D) Probar el software.

Referencia: Actividades sombrilla en el desarrollo de software.

26. ¿Qué actividad sombrilla permite que el equipo de software monitoree el progreso del proyecto?

- A) **Seguimiento y control del proyecto.**
- B) Aseguramiento de la calidad del software.
- C) Gestión de la reutilización.
- D) Modelado.

Referencia: Actividades sombrilla en el desarrollo de software.

27. ¿Qué implica el principio de "Generalidad" en ingeniería de software?

- A) Crear soluciones específicas para cada problema.
- B) **Buscar soluciones reutilizables.**
- C) Eliminar redundancias en el código.
- D) Incrementar el tamaño del código.

Referencia: Principios esenciales de ingeniería de software

28. ¿Cuál es el propósito del "End User" en un proyecto de software?

- A) Gestionar el proyecto.
- B) Realizar las pruebas de sistema.
- C) **Consumir el producto final.** D) Diseñar el sistema.

Referencia: Tabla de roles y responsabilidades.

29. ¿Cuál es el enfoque principal de la "Resolución de problemas" en ingeniería de software?

- A) Diseñar software sin requerimientos específicos.
- B) **Resolver un problema específico planteado por el cliente.**
- C) Eliminar la necesidad de cambios futuros.
- D) Minimizar el costo de desarrollo.

Referencia: Descripción de la resolución de problemas en ingeniería de software.

30. ¿Cuál de los siguientes autores define la ingeniería de software como "el estudio de los principios y metodologías para desarrollo y mantenimiento de sistemas de software"?

- A) Bauer (1972).
- B) Boehm (1976).
- C) **Zelkovitz (1978).**
- D) Pressman (2005)

Referencia: Definiciones de ingeniería de software.

31. ¿Cuál es la principal diferencia entre la ingeniería de software y otras disciplinas de ingeniería como la civil o eléctrica?

- A) La ingeniería de software utiliza menos recursos.
- B) **La ingeniería de software se basa en el conocimiento como "materia prima".**
- C) La ingeniería de software no requiere diseño.
- D) La ingeniería de software es tangible

Referencia: "la 'materia prima' es el conocimiento" (Naturaleza Intangible del Producto y Relación con Otras Disciplinas de Ingeniería).

32. ¿Qué característica tiene el software una vez que se desarrolla?

- A) Se desgasta físicamente con el tiempo.
- B) Es difícil de replicar.
- C) **Puede replicarse de manera exacta y a bajo costo.**
- D) Necesita procesos de manufactura adicionales.

Referencia: "se puede replicar de manera exacta y a bajo costo" (Producción y Manufactura).

33. ¿A qué está propenso el software, a pesar de no experimentar desgaste físico?

- A) Al deterioro físico.
- B) **A la obsolescencia tecnológica.**
- C) A una vida útil infinita.
- D) A una mejora continua sin fallos

Referencia: "el software es susceptible a la obsolescencia" (Producción y Manufactura).

34. ¿Qué teoría ayuda a entender la evolución del software?

- A) Leyes de la Física de Newton.
- B) Leyes de la Electrónica.
- C) **Leyes de la Evolución del Software de Lehman.**
- D) Leyes de la Computación de Turing

Referencia: "las 'Leyes de la Evolución del Software' de Lehman" (Enfoque Científico en la Ingeniería de Software).

35. ¿Cómo puede "deteriorarse" el software a diferencia de productos físicos?

- A) A través del desgaste por uso.
- B) **Por la falta de actualizaciones y fallos acumulados.**
- C) Por su resistencia limitada.
- D) Por la pérdida de recursos tangibles.

Referencia: "puede 'deteriorarse' de manera conceptual debido a la acumulación de fallos o a la falta de compatibilidad con nuevos entornos operativos" (Mantenimiento y Evolución).

36. ¿Qué área relacionada se centra en teorías y algoritmos, a diferencia de la ingeniería de software?

- A) Ingeniería de sistemas.
- B) Ingeniería eléctrica.
- C) **Ciencias de la computación.**
- D) Matemáticas aplicadas

Referencia: "la ciencia de la computación, que se enfoca en teorías y algoritmos" (Interacción con Otras Disciplinas de la Computación y Ciencias).

37. ¿Cuál de las siguientes organizaciones es reconocida por promover estándares de calidad en la ingeniería de software?

- A) WHO.
- B) IEEE.**
- C) NASA.
- D) UNICEF

Referencia: "organizaciones acreditadas, como ACM e IEEE" (La Ingeniería de Software como Profesión).

38. ¿Cuál de los siguientes es un desafío distintivo en el ciclo de vida del software en comparación con otras ingenierías?

- A) Su duración indefinida.
- B) La dificultad para replicarlo.
- C) El rápido ciclo de vida por cambios tecnológicos.**
- D) La estabilidad y consistencia en el tiempo

Referencia: "el corto ciclo de vida de sus productos" (Obsolescencia y Rápido Ciclo de Vida).

39. ¿Qué parte de otras ingenierías ha sido adaptada para la ingeniería de software?

- A) Manufactura en línea.
- B) Uso de materiales tangibles.
- C) Gestión de proyectos y modelado estructurado.**
- D) Construcción de puentes

Referencia: "la gestión de proyectos en ingeniería civil y el modelado estructurado se han adaptado" (Adaptación de Conocimientos y Metodologías de Otras Ingenierías).

40. ¿Cuál es un aspecto en el que la ingeniería de software y la ingeniería de sistemas se diferencian?

- A) La ingeniería de sistemas abarca tanto hardware como software.**
- B) La ingeniería de sistemas es solo teoría.
- C) La ingeniería de software no trabaja con sistemas complejos.
- D) La ingeniería de software solo se enfoca en hardware

Referencia: "la ingeniería de sistemas... integra tanto hardware como software" (Interacción con Otras Disciplinas de la Computación y Ciencias).

41. ¿Qué aspecto ético implica proteger los datos sensibles de los usuarios en ingeniería de software?

- A) Transparencia.
- B) Competencia Profesional.
- C) **Privacidad.**
- D) Sostenibilidad

Referencia: Video del tutorial

42. ¿Cuál es uno de los principios clave en los métodos ágiles?

- A) Desarrollo en fases finales.
- B) **Entrega incremental de funcionalidades.**
- C) Análisis detallado previo al desarrollo.
- D) Revisión anual del producto

Referencia: Video del tutorial

43. ¿Qué tipo de requerimiento describe la “velocidad de respuesta” en un sistema?

- A) Funcional
- B) Estético
- C) **No funcional**
- D) Operativo

Referencia: Video del tutorial

44. ¿Cuál de los siguientes patrones arquitectónicos organiza el sistema en capas que se pueden modificar de manera independiente?

- A) Cliente-Servidor.
- B) Microservicios.
- C) Modelo-Vista-Controlador (MVC).
- D) **Arquitectura en capas**

Referencia: Video del tutorial

45. ¿Cuál de los siguientes atributos evalúa la capacidad del software para funcionar sin errores?

- A) Usabilidad.
- B) **Fiabilidad.**
- C) Funcionalidad.

D) Eficiencia.

Referencia: Video del tutorial

46. ¿Qué tipo de prueba evalúa la usabilidad del sistema desde la perspectiva del usuario?

A) Pruebas de integración.

B) Pruebas de sistema.

C) Pruebas unitarias.

D) **Pruebas de usuario**

Referencia: Video del tutorial

47. ¿Cuál es el propósito principal de la administración del cambio en el desarrollo de software?

A) Crear nuevos requerimientos.

B) **Documentar y controlar cada modificación.**

C) Reescribir el código base.

D) Reducir el tiempo de desarrollo.

Referencia: Video del tutorial

48. ¿Qué herramienta se utiliza para rastrear y restaurar versiones anteriores del código?

A) Gestión de requerimientos.

B) **Control de versiones.**

C) Pruebas de calidad.

D) Administración de proyectos

Referencia: Video del tutorial

49. ¿Cuál es uno de los métodos ágiles que organiza el trabajo en "sprints"?

- A) Scrum.**
- B) XP.
- C) Modelo en cascada.
- D) Lean

Referencia: Video del tutorial

50. ¿Qué atributo de calidad del software se enfoca en el uso óptimo de recursos?

- A) Mantenibilidad.
- B) Fiabilidad.
- C) **Eficiencia.**
- D) Funcionalidad.

Referencia: Video del tutorial